

# 基于最大熵的隐马尔可夫模型文本信息抽取

林亚平, 刘云中, 周顺先, 陈治平, 蔡立军

(湖南大学计算机与通信学院, 湖南长沙 410082)

**摘 要:** 文本信息抽取是处理海量文本的重要手段之一. 最大熵模型提供了一种自然语言处理的方法. 提出了一种基于最大熵的隐马尔可夫模型文本信息抽取算法. 该算法结合最大熵模型在处理规则知识上的优势, 以及隐马尔可夫模型在序列处理和统计学习上的技术基础, 将每个观察文本单元所有特征的加权之和用来调整隐马尔可夫模型中的转移概率参数, 实现文本信息抽取. 实验结果表明, 新的算法在精确度和召回率指标上比简单隐马尔可夫模型具有更好的性能.

**关键词:** 人工智能; 文本信息抽取; 隐马尔可夫模型; 最大熵

**中图分类号:** TP391 **文献标识码:** A **文章编号:** 0372-2112 (2005) 02-0236-05

## Using Hidden Markov Model for Text Information Extraction Based on Maximum Entropy

LIN Ya-ping, LIU Yun-zhong, ZHOU Shun-xian, CHEN Zhi-ping, CAI Li-jun

(College of Computer and Communication, Hunan University, Changsha, Hunan 410082, China)

**Abstract:** Text information extraction is an important approach to processing large quantity of text. Maximum entropy provides a kind of framework for natural language processing. A new algorithm using hidden Markov model based on maximal entropy is proposed for text information extraction. The new algorithm combines the advantage of maximum entropy model, which can integrate and process rules and knowledge efficiently, with that of hidden Markov model, which has powerful technique foundations to solve sequence representation and statistical problem, and uses the sum of all features with weights to adjust the transition parameters in hidden Markov model for text information extraction. Experimental results show that the new algorithm improves the performance in precision and recall.

**Key words:** artificial intelligence; text information extraction; hidden markov model; maximal entropy

### 1 引言

WWW的普遍应用导致网上的文本数量成指数级增长, 因此如何自动处理这些海量联机文本信息成为目前重要的研究课题. 自动文本信息抽取是文本信息处理的一个重要环节<sup>[1]</sup>. 文本信息抽取(text information extraction)是指从文本中自动抽取相关的或特定类型的信息. 目前文本信息抽取模型主要有三种: 基于词典的抽取模型<sup>[2]</sup>, 基于规则的抽取模型<sup>[3]</sup>和基于隐马尔可夫模型(Hidden Markov Model, HMM)的抽取模型<sup>[4~8]</sup>.

利用HMM进行文本信息抽取是一种基于统计机器学习的信息抽取方法. HMM易于建立, 不需大规模的词典集与规则集, 适应性好, 抽取精度较高, 因而得到研究者的关注. 文献[4]应用HMM抽取计算机科学研究论文的标题、作者和摘要等头部信息; 文献[5]使用一种“收缩(shrinkage)”的技术改进HMM信息抽取模型概率的估计; 文献[6]使用随机优化技术

动态选择最适合的HMM模型结构进行信息抽取; 文献[7]将自然语言处理中的短语结构分析技术应用到HMM文本信息抽取中; 文献[8]利用主动学习技术减少训练HMM信息抽取模型时所需的标记数据. 采用HMM存在的主要问题是没有考虑文本的上下文特征信息和文本词汇本身包含的特征信息, 而这些信息对于实现正确的文本信息抽取是非常有用的. 文献[9]提出最大熵的马尔可夫模型(Maximum entropy Markov model, MEMM), 用来对FAQs文本中的问题和答案部分进行切分. MEMM是一种指数模型, 主要是将文本的抽象特征作为输入, 并在马尔可夫状态转移的基础上选择下一个状态, 这一点上其更接近于有穷状态自动机. 由于MEMM将文本的上下文特征信息以及文本词汇本身包含的特征信息结合到马尔可夫模型中来, 可以改善信息抽取的性能. 但它没有对具体的文本词汇进行统计, 只考虑了抽象特征, 这也导致其在某些情况下性能不如HMM. 本文结合最大熵模型在处理规则知识表示方面的优势, 以及隐马尔可夫模型在序列处理和统计学习上的

技术基础,提出一种基于最大熵的隐马尔可夫模型(Maximum entropy based hidden Markov model, ME-HMM),实现文本信息抽取.该算法将每个观察文本单元所有特征的加权之和用来调整隐马尔可夫模型中的转移概率参数.实验结果表明,新的算法在精确度和召回率指标上比隐马尔可夫模型具有更好的性能.

## 2 基于 HMM 的文本信息抽取

### 2.1 HMM 模型

HMM 提供了一种基于训练数据的概率自动构造识别系统的技术.一个 HMM 包含两层:一个可观察层和一个隐藏层,可观察层是待识别的观察序列,隐藏层是一个马尔可夫过程,即是一个有限状态机,其中每个状态转移都带有转移概率<sup>[10]</sup>.

一个 HMM 可以看成是一个五元组  $\{S, V, A, B, \pi\}$ :

$$S = \{S_1, S_2, \dots, S_N\}$$

$$V = \{V_1, V_2, \dots, V_M\}$$

$$A = \{a_{ij} = P(q_{t+1} = S_j | q_t = S_i), 1 \leq i, j \leq N\}$$

$$B = \{b_j(V_k) = P(V_k \text{ at } t | q_t = S_j), 1 \leq j \leq N, 1 \leq k \leq M\}$$

$$\pi = \{\pi_i = P(q_1 = S_i), 1 \leq i \leq N\}$$

其中  $S$  是状态集,共有  $N$  个状态; $V$  是词汇集,共有  $M$  个可能的输出单词; $A$  是  $N \times N$  的状态转移矩阵, $a_{ij}$  表示从状态  $S_i$  转换到状态  $S_j$  的概率; $B$  是  $N \times M$  的释放概率矩阵, $b_j(V_k)$  表示在状态  $S_j$  时释放单词  $V_k$  的概率; $\pi$  是初始状态概率集合, $\pi_i$  是第  $i$  个状态作为初始状态的概率.

### 2.2 基于 HMM 的文本信息抽取

应用 HMM 模型,主要解决以下三个方面的问题:评估问题、学习问题和解码问题.常用的算法参见文献[10].文本信息抽取需要解决 HMM 模型中的学习问题和解码问题.进行信息抽取时,一般是先训练样本,采用 ML(Maximum likelihood)算法(对于已标记训练样本)或 Baumr Welch 算法(对于未标记训练样本)进行学习,得出 HMM 模型参数.然后采用 Viterbi 算法将待抽取的输入文本序列标记为最大概率的状态标签序列.状态标签是我们事先定义的待抽取内容标签.本文基于 HMM 模型的文本信息抽取采用已手工标记的训练样本训练 HMM 模型,信息抽取分为两大步骤:

(1)应用统计的方法从训练样本中得出 HMM 模型参数.采用 ML 算法,建立 HMM 模型.ML 算法主要以统计的方法得出 HMM 模型参数,由以下三个公式分别计算模型的初始状态概率、转移状态概率和状态释放概率,即

$$\pi_i = \frac{\text{Init}(i)}{\sum_{j=1}^N \text{Init}(j)}, 1 \leq i \leq N \quad (1)$$

其中,  $\text{Init}(i)$  是所有训练序列中,初始状态为  $i$  的序列个数.

$$a_{ij} = \frac{C_{i,j}}{\sum_{k=1}^N C_{i,k}}, 1 \leq i, j \leq N \quad (2)$$

其中,  $C_{i,j}$  是所有训练序列中,从状态  $S_i$  转换到状态  $S_j$  的次数.

$$b_j(V_k) = \frac{E_j(V_k)}{\sum_{i=1}^M E_j(V_i)}, 1 \leq j \leq N, 1 \leq k \leq M \quad (3)$$

其中,  $E_j(V_k)$  是所有训练序列中,状态  $S_j$  释放单词  $V_k$  的次数.

(2)应用已建立好的 HMM 模型进行文本信息抽取.以文本观察序列  $O = O_1 O_2, \dots, O_T$  作为模型输入,采用 Viterbi 算法,找出最大概率的状态标签序列,被标记为目标状态标签的观察文本即为信息抽取的内容.

## 3 基于最大熵的隐马尔可夫模型文本信息抽取算法

### 3.1 最大熵原理

最大熵原理是热力学中非常重要的一个原理,在许多其他领域有着广泛的应用,在自然语言处理方面也是一种主要的处理方法<sup>[11]</sup>.

若将自然语言看作一个随机过程,建立随机过程的模型  $p, P$ ; 输出值集合  $Y, y$ ; 上下文集合  $X, x$ ;  $N$  个样本的集合  $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}, (x_i, y_i)$  是观察到的一个事件,事件空间  $X \times Y$ ; 语言知识用特征表示,特征是一个 2 值函数  $f: X \times Y \rightarrow \{0, 1\}$ .熵刻画了随机变量的不确定性.随机变量的不确定性越大,它的熵越大.

模型  $p$  的熵:

$$H(p) = - \sum_{x,y} p(x,y) \log p(x,y) \quad (4)$$

最大熵模型:

$$p^* = \arg \max_p H(p) \quad (5)$$

$C$  是满足约束的模型集合,剩下的问题就是在  $C$  中寻找  $p^*$ ,

$p^*$  具有下面的形式:

$$p^*(y|x) = \frac{1}{Z(x)} \exp \left( \sum_i f_i(x,y) \right) \quad (6)$$

其中  $Z(x)$  是一个归一化常量.

$$Z(x) = \sum_y \exp \left( \sum_i f_i(x,y) \right) \quad (7)$$

$i$  是模型的参数,也可以看作是特征的权值. Darroch 和 Ratcliff 于 1972 年提出一个称为 GIS (generalized iterative scaling algorithm) 的算法,该算法是一般的迭代算法<sup>[12]</sup>.最大熵方法的一个好处是,实验者仅仅只需要将精力集中在选择什么样的特征上,而不用考虑如何使用这些特征.每个特征的贡献由相应的权值来决定,而这些权值可由 GIS 学习算法自动得到.

### 3.2 基于最大熵的隐马尔可夫模型文本信息抽取算法

在最大熵方法中,我们称规则为特征,最大熵方法就是找出一个特征集合,并确定每条特征的重要程度.最大熵模型能集成各种特征与规则到一个统一的框架下,而隐马尔可夫模型在处理序列问题和统计方面有着很强的优势,将最大熵方法结合到隐马尔可夫模型文本信息抽取中来,既能解决知识表示问题,又可以随时把新获取的语言知识添加到模型中去,二者的结合是规则方法与统计方法的结合.依据这一思想,本文基于最大熵原理,结合隐马尔可夫模型进行文本信息抽取.为了便于比较,图 1 分别给出了 HMM 模型、MEMM 模型和本

文提出的 ME-HMM 模型.

给定特征集,定义一个二值函数:

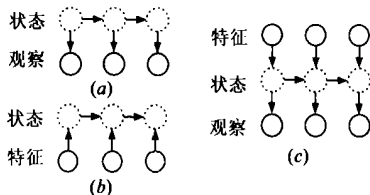


图 1 (a)HMM;(b)MEMM;(c)ME-HMM

$$f_{i,j}(o_t, s_t) = \begin{cases} 1, & \text{若观察 } o_t \text{ 具有该特征 } i, \text{ 且标记状态 } s_t = s_j \\ 0, & \text{否则} \end{cases} \quad (8)$$

ME-HMM 模型建立一个特征-状态转移概率矩阵  $M = \{M_{i,j}\}$ . 其中矩阵单元  $M_{i,j}$  为从特征  $i$  到状态  $j$  的转移概率, 满足下面的条件:

$$M_{i,j} = 1, \quad 1 \leq i \leq N_F, 1 \leq j \leq N_S \quad (9)$$

上面公式中  $N_F$  是选定特征个数,  $N_S$  是模型状态个数. 在训练阶段, 对各个观察进行特征提取, 而训练数据集中的每一个观察隐含对应于一个标记状态, 因此可以统计出状态与特征之间的关系. 由 GIS 算法<sup>[13]</sup>可以得到特征-状态转移概率矩阵. 算法如下:

(1) 计算训练数据中每个特征-状态的平均值.

观察序列总长度为  $m_s$  时, 对第  $i$  个特征, 第  $j$  个状态, 其平均值为

$$F_{i,j} = \frac{1}{m_s} \sum_{k=1}^{m_s} f_{i,j}(o_k, s_k) \quad (10)$$

(2) 以任意的参数开始作为 GIS 算法的第 0 次迭代, 可以令  $M_{i,j}^{(0)} = 1$ .

(3) 在第  $n$  次迭代, 使用当前的  $M_{i,j}^{(n)}$  值, 计算每个特征-状态的期望值

$$E_{i,j}^{(n)} = \frac{1}{m_s} \sum_{k=1}^{m_s} \sum_s P_s^{(n)}(s | o_k) f_{i,j}(o_k, s) \quad (11)$$

其中,  $s$  是  $s$  上一个状态, 这里,

$$P_s^{(n)}(s | o_k) = \frac{1}{Z(o, s)} \exp \left( \sum_i M_{i,j}^{(n)} f_{i,j}(o_k, s) \right) \quad (12)$$

其中,  $Z(o, s)$  是归一化常数, 保证对所有  $s$  状态的所有下一个状态  $s$  加和为一.

(4) 在满足约束条件限制的情况下, 通过将期望值和训练数据平均值作比较, 以决定该怎样调整参数. 选定一常数  $C$ , 调整公式如下.

$$M_{i,j}^{(n+1)} = M_{i,j}^{(n)} + \frac{1}{C} \log \left( \frac{F_{i,j}}{E_{i,j}^{(n)}} \right) \quad (13)$$

(5) 如果参数值达到收敛, 则结束算法, 否则, 转到步骤 (3).

为了减少迭代次数, 我们使用统计的方法为  $M_{i,j}^{(0)}$  赋初值. 在应用 Viterbi 算法时,  $t$  时刻的状态由  $t-1$  时刻的状态和  $t$  时刻的观察特征共同依概率决定.

$$P(s_t = s_j | s_{t-1}, o_t) = \frac{1}{\sum_i (t-1, j + (1 - ) * }$$

$$(M_{i,j} * f_{i,j}(o_t, s_t)) \quad (14)$$

其中,  $t-1, j$  是已知的  $t-1$  时刻的状态到状态  $j$  的转移概率, 为归一化参数.

$$= \frac{1}{\sum_i (t-1, j + (1 - ) * (M_{i,j} * f_{i,j}(o_t, s_t)) \quad (15)$$

是调整特征-状态转移概率与状态转移概率的相对重要程度的权重.

## 4 实验

为了便于比较, 本文利用美国卡耐基梅隆 (CMU) 大学提供的两个标准数据集进行了实验, 一个是计算机科研论文头部信息抽取数据集, 另一个是学术报告通知中关键信息的抽取数据集. 为了降低最大熵方法中的特征选择过程的时间复杂度, 我们人为地选定一些有用的特征. 比如, 在从学术报告通知中抽取关键信息时我们主要使用以下一些特征:

表 1 学术报告通知中文本信息抽取中所使用的一些特征

1. 是否 time 型	2. 是否人名
3. 是否以 "Time:" 作为上下文的头	4. 是否以 "PostedBy:" 作为上下文的头
5. 是否句首的第一个词	6. 是否以 "speaker" 作为上下文的头
7. 是否以 "WHO:" 作为上下文的头	8. 是否以 "author:" 作为上下文的头
9. 是否以 "by:" 作为上下文的头	10. 是否以 "LOCATION" 作为上下文的头
11. 是否以大写字母开头	12. 是否全部是数字
13. 是否以 "Place:" 作为上下文的头	14. 是否以 "discussaut:" 作为上下文的头
15. 是否以 "present:" 作为上下文的头	16. 是否以 "panelists:" 作为上下文的头

针对具体的状态, 根据特征属性相对于该状态的贡献, 可分为正特征和负特征. 状态的正特征表示当观察出现该特征时, 倾向于转移到该状态; 状态的负特征表示当观察出现该特征时, 转移到该状态的概率会降低. 在判断是否包含人名特征时, 我们使用从网上下载的美国人名字典进行匹配.

### 4.1 计算机科研论文头部信息抽取

从计算机科研论文头部抽取信息时, 考虑到排版格式、换行、分隔符等有用的信息. 需要先进行文本预处理, 采用基于文本分块的 HMM 信息抽取. 在训练阶段, 初始概率和转移概率以分块为基本单位, 公式 (1) 和 (2) 被使用; 释放概率以单词为基本单位, 使用公式 (3). 在抽取阶段, 将待抽取的文本进行分块后, 计算各分块的释放概率. 待抽取文本序列转换为分块序列, 块释放概率为块内各单词的释放概率之和. 设块观察序列为  $O = O_1 O_2 \dots O_T$ , 若第  $t$  块的长度为  $K$ , 即包含有  $K$  个单词, 记为:  $O_{t1} O_{t2} \dots O_{tK}$ , 则状态  $j$  释放第  $t$  块的概率为:

Data set for information extraction, <http://www2.cs.cmu.edu/~kseymore/ie.html>

RISE: Repository of Online Information Sources Used in Information Extraction Tasks, <http://www.isi.edu/info-agents/RISE/>  
<http://www.census.gov/genealogy/names>

$$b_j(O_i) = \frac{K}{k=1} b_j(O_{ik}) \quad (16)$$

我们仅使用比较少的几个特征,像是否人名、是否全为数字、是否月份或其缩写、是否包含 email 的标志“@”(不包含该标志的分块一定不是 email)等等。实验中,通过改变参数值,来调整特征-状态转移概率与状态转移概率的相对重要程度。当  $\alpha = 0.6$  时,实验结果见图 2 和表 2。

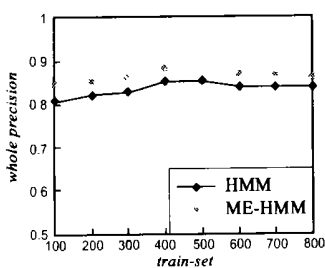


图 2 HMM 与 ME-HMM 信息抽取总精度比较

表 2 HMM 与 ME-HMM 信息抽取具体域精确度和召回率比较 (train-set = 400)

	HMM		ME-HMM	
	Precision	Recall	Precision	Recall
Title	0.820160	0.831591	0.839409	0.865979
Author	0.823954	0.916533	0.898551	0.995185
Affiliation	0.873176	0.915294	0.887372	0.917647
Address	0.903797	0.840000	0.926020	0.854118
Email	0.888298	1.000000	0.989547	1.000000
Note	0.880795	0.716338	0.922434	0.693896
Web	1.000000	0.547619	1.000000	0.547619
Phone	0.986667	0.888889	0.986301	0.913580
Date	0.660606	0.990909	0.698718	0.990909
Abstract	0.954545	1.000000	0.910082	1.000000
Intro	0.873418	1.000000	0.873418	1.000000
Keyword	0.825279	0.932773	0.825279	0.932773
Degree	0.513514	0.838235	0.617647	0.813043
Pubnum	0.885714	0.643939	0.955056	0.704545
Page	1.000000	1.000000	1.000000	1.000000

如果不考虑词汇的统计概率,使用 MEMM 进行文本信息抽取,在本文实验数据集上性能比较差,不便于比较,所以我们没有列出相关的实验结果。从图 2 和表 2 可以看出,基于最大熵的隐马尔可夫模型文本信息抽取算法在精确度和召回率上有一定的提高。图 2 中随着训练文本的增加,总精确度有所下降,这是因为增加的训练集与测试集不是很匹配所造成的。当使用人名词典进行匹配时,表 2 显示“author”状态的召回率提高了约 8%,这说明相关知识的加入起了很大作用。

#### 4.2 学术报告通知中关键信息抽取

在对计算机科研论文头部信息进行抽取时,其实就是对文本中所有分块进行分类标记的过程。为了能从大量文本中抽取少量的关键信息,我们以学术报告通知中关键信息抽取作为研究对象。我们需要从学术报告通知中抽取报告的起始时间、结束时间、报告人以及报告地点等信息<sup>[5,6]</sup>。

为了从学术报告通知中抽取关键信息,模型主要由四个主要状态组成:前缀状态、后缀状态、背景状态和目标状态。前缀状态和后缀状态可能有几个。根据被抽取信息的长度,目标状态也可长可短。多条目标状态路径也被使用,在稀疏训练数据情况下,“收缩”技术被使用来计算释放概率。图 3 分别是由

四种主要状态组成的 HMM 模型多路径拓扑及非目标状态收缩配置示意图。

对于每一个观察,在每一层结点上都有一个释放概率,假设对于单个观察词汇  $w$  的释放概率集合为:  $\{P(w|s_j^0), P(w|s_j^1), \dots, P(w|s_j^k)\}$ 。  $P(w|s_j^0)$  是最低层结点的释放概率,  $P(w|s_j^k)$  是最顶层结点的释放概率。

$$P(w|s_j) = \sum_{i=0}^k \alpha_i * P(w|s_j^i) \quad (17)$$

其中,  $\alpha_i$  是相应的释放概率所占的权重,并满足条件:

$$\sum_{i=0}^k \alpha_i = 1 \quad (18)$$

该权重由 EM 算法学习得到<sup>[5]</sup>。

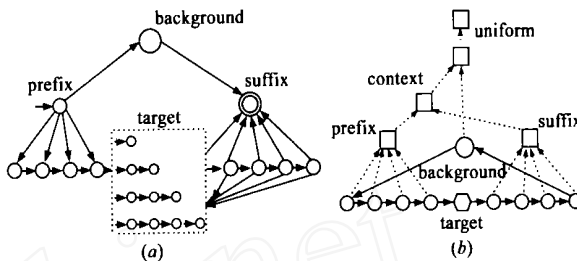


图 3 (a)HMM 模型多路径拓扑图; (b) 非目标状态的收缩配置图 (虚箭头表示收缩方向)

我们使用 CMU 大学提供的 485 个学术报告通知文件进行实验,使用表 1 中所列出的特征,交叉测试,取平均值,当  $\alpha = 0.5$  时,实验结果见表 3。从表 3 可以看出,在四个域具体的文本信息抽取实验中,数据的平均值表明使用最大熵的隐马尔可夫模型在精确度和召回率上比基于简单的隐马尔可夫模型要高。由于使用了上下文信息,如表 1 所列出的特征,其性能改善得比较明显,而一阶的隐马尔可夫模型是不可能利用到这些信息的。但另一方面我们也发现,当特征很多时,多个特征之间重迭时,参数学习问题变得很困难。

表 3 学术报告通知文件中信息抽取精确度和召回率比较

	HMM		ME-HMM	
	Precision	Recall	Precision	Recall
Speaker	0.821249	0.756757	0.827957	0.901961
Location	0.885714	0.763158	0.925926	0.828571
Stime	0.966667	0.783784	0.968750	0.830769
Etime	0.647059	0.687500	0.846154	0.758621

另外,表 3 中的数据显示有时候召回率有很大的提高,但精确度并没提高。

## 5 小结

最大熵模型提供了一种自然语言处理的方法,能集成各种特征与规则到一个统一的框架下。将最大熵方法结合到隐马尔可夫模型文本信息抽取中来,解决了知识表示的问题,可以随时把新获取的语言知识添加到模型中去,实验结果也表明了该方法是有效的。在后续工作中我们打算让系统自动从很多候选集中筛选特征,另外,如何降低算法复杂度也值得进一步的研究。

## 参考文献:

- [1] S Lawrence, L Giles, K Bollacker. Digital libraries and autonomous citation indexing[J]. IEEE Computer, 1999, 32(6): 67 - 71.
- [2] E Riloff, R Jones. Learning dictionaries for information extraction by multi-level bootstrapping[A]. Proceedings of the Sixteenth National Conference on Artificial Intelligence[C]. Orlando: AAAI Press, 1999. 811 - 816.
- [3] Kushmerick N. Wrapper induction: Efficiency and expressiveness[J]. Artificial Intelligence, 2000, 118(12): 15 - 68.
- [4] Kristie Seymore, Andrew McCallum, Ronal Rosenfel. Learning hidden Markov model structure for information extraction[A]. Proceedings of the AAAI '99 Workshop on Machine Learning for Information Extraction[C]. Orlando: AAAI Press, 1999. 37 - 42.
- [5] Dayne Freitag, Andrew McCallum. Information extraction with HMMs and shrinkage[A]. Proceedings of the AAAI '99 Workshop on Machine Learning for Information Extraction[C]. Orlando: AAAI Press, 1999. 31 - 36.
- [6] Freitag D, McCallum A. Information extraction with HMM structures learned by stochastic optimization[A]. Proceedings of the Eighteenth Conference on Artificial Intelligence[C]. Edmonton: AAAI Press, 2002. 584 - 589.
- [7] Souyma Ray, Mark Craven. Representing sentence structure in hidden Markov models for information extraction[A]. Proceedings of the Seventeenth International Joint Conference On Artificial Intelligence[C]. Washington: Morgan Kaufmann, 2001. 1273 - 1279.
- [8] T Scheffer, C Decomain, S Wrobel. Active hidden Markov models for

information extraction[A]. Proceedings of the Fourth International Symposium on Intelligent Data Analysis[C]. Lisbon: Springer, 2001. 301 - 109.

- [9] Freitag D, McCallum A, Pereira F. Maximum entropy Markov models for information extraction and segmentation[A]. Proceedings of The Seventeenth International Conference on Machine Learning[C]. San Francisco: Morgan Kaufmann, 2000. 591 - 598.
- [10] Lawrence E Rabiner. A tutorial on hidden Markov models and selected application in speech recognition[J]. Proceedings of the IEEE, 1989, 77(2): 257 - 286.
- [11] Adam L Berger, Stephen A Della Pietra, Vincent J Della Pietra. A maximum entropy approach to natural language processing[J]. Computational Linguistics, 1996, 22(1): 39 - 71.
- [12] Darroch J, Ratcli D. Generalized iterative scaling for log-linear models[J]. Annals of Mathematical Statistics, 1972, 43(5): 1470 - 1480.

## 作者简介:



林亚平 男, 1955 年出生于湖南邵阳, 湖南大学计算机与通信学院教授, 博士生导师, 主要研究方向为计算机通信网络, 机器学习. E-mail: yplin@hnu.edu.cn.

刘云中 男, 1980 出生于湖北大冶, 湖南大学计算机与通信学院硕士研究生, 主要研究方向为机器学习.